Making Chatbot with Dialogflow

Dialogflow


IVERSON
LEADER
IN PROFESSIONAL
IT TRAINING
<code/>

Agenda

- What is Chatbot
- Chatbot Architecture
- Chatbot Platform / Bot Engine
- Making Chatbot (for FAQ)
- Making Chatbot (with Node.js)
- Publish Chatbot in Different Channels

Presenter

/arifmazumder

Mohammed Arif, PhD

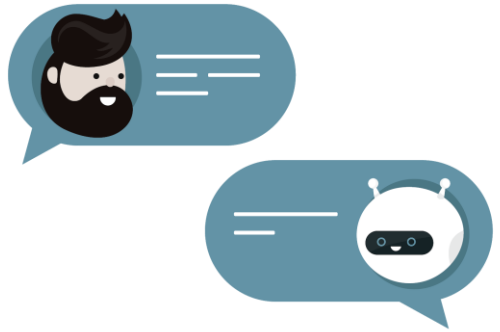
Senior Technical Consultant

Big Data | Machine Learning | AI

Mohammed Arif has more than twelve (13) years of working experience in Information Communication and Technology (ICT) industry. The highlights of his career are more than six (7) years of holding various senior management and/or C-Level and had five (5) years of international ICT consultancy exposure in various countries (APAC and Australia), specially on Big Data, Data Engineering, Machine Learning and AI arena.

He is also Certified Trainer for Microsoft.

What is ChatBot



A chatbot is an artificial intelligence (AI) software(Agent) that can simulate a conversation (or a chat) with a user in natural language (NLP) through messaging applications, websites, mobile apps or through the telephone (Channel).

How ChatBot Works

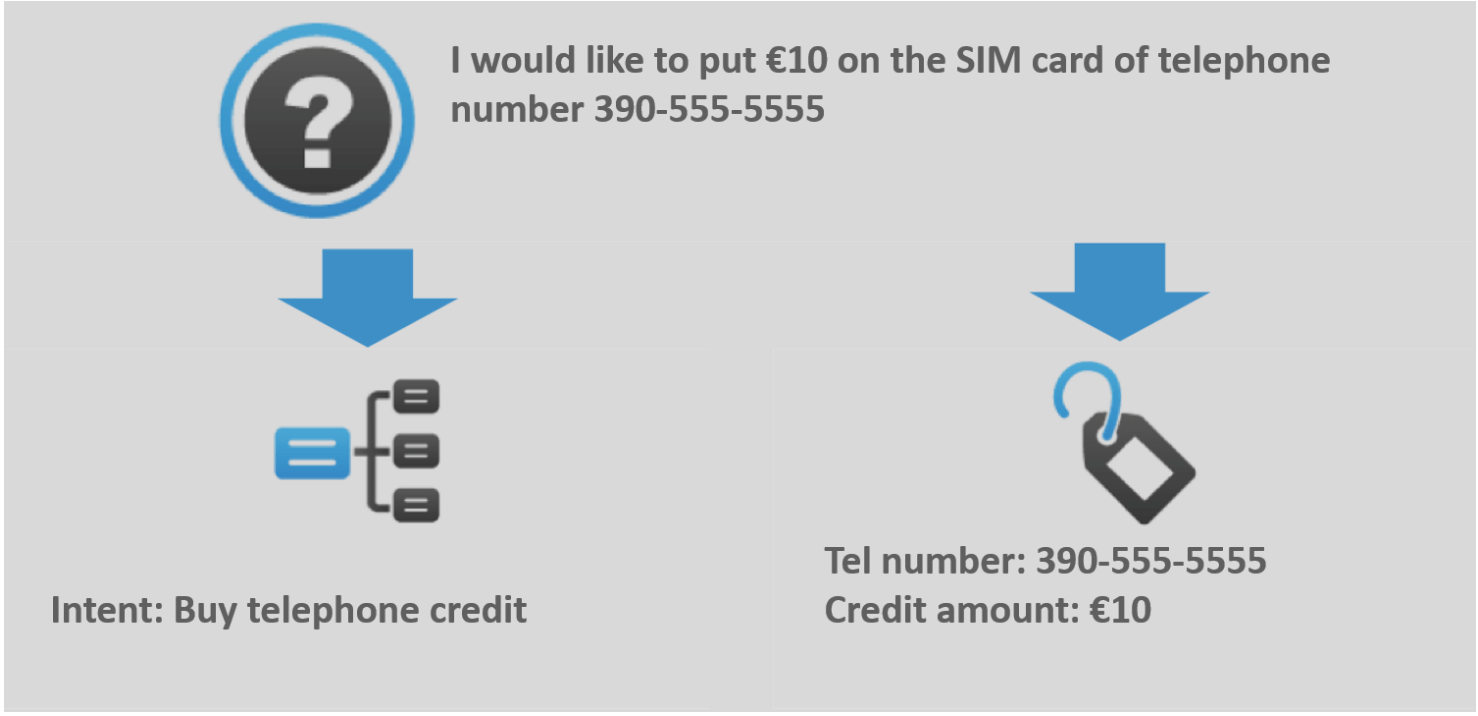
There are two different tasks at the core of a chatbot:

1. User request analysis
2. Returning the response

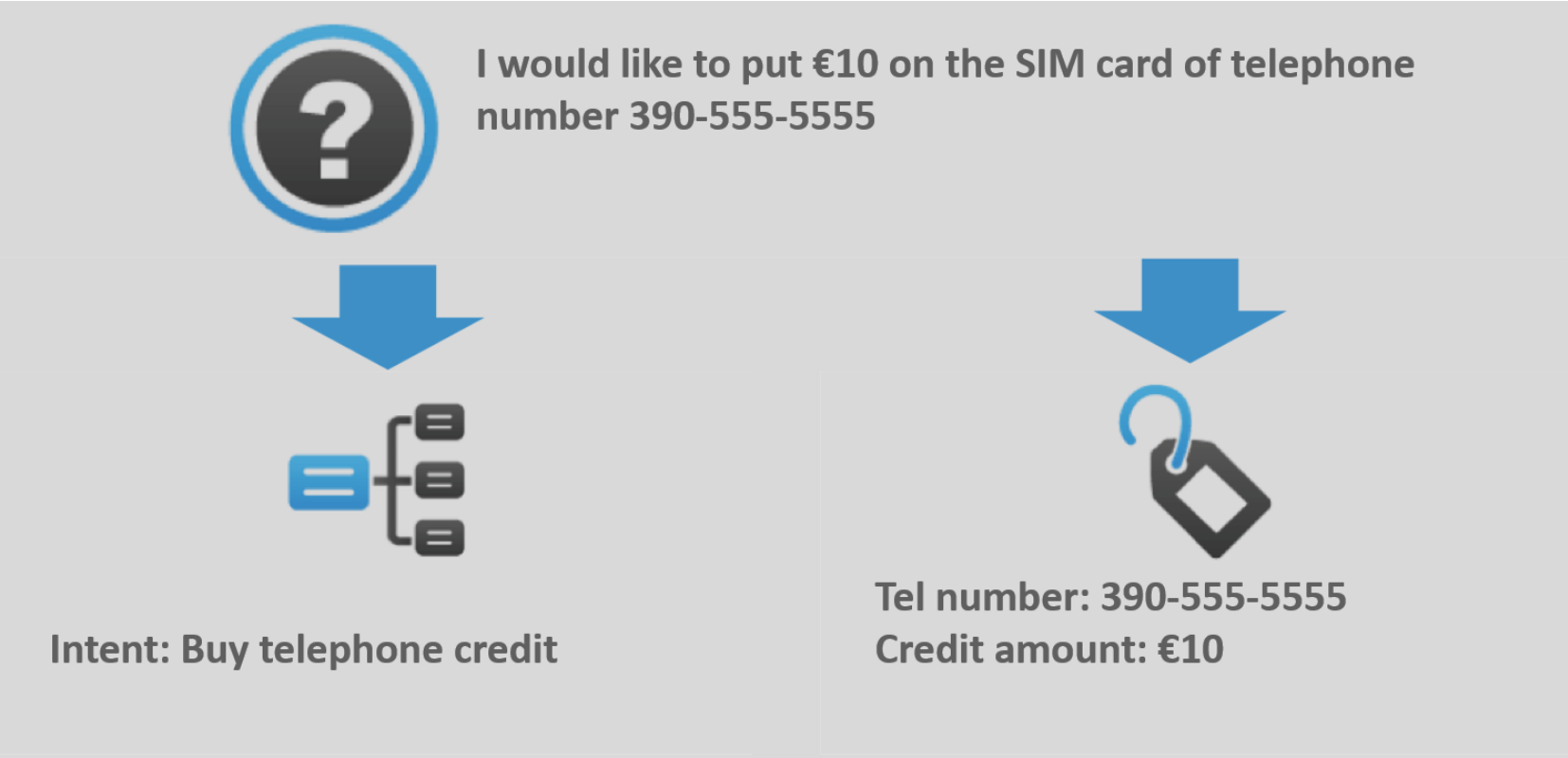
How ChatBot Works

1. User request analysis:

   This is the first task that a chatbot performs. It analyzes the user's request to identify the user intent and to extract relevant entities.



How ChatBot Works



If you are not able to correctly understand the user's request, you won't be able to provide the correct answer.

The ability to identify the user's intent and extract data and relevant entities contained in the user's request is the first condition and the most relevant step at the core of a chatbot.

How ChatBot Works

2. Returning the response:

Once the user's intent has been identified, the chatbot must provide the most appropriate response for the user's request. The answer may be:

::: a generic and predefined text

::: a text retrieved from a knowledge base that contains different answers

::: a contextualized piece of information based on data the user has provided

::: data stored in enterprise systems

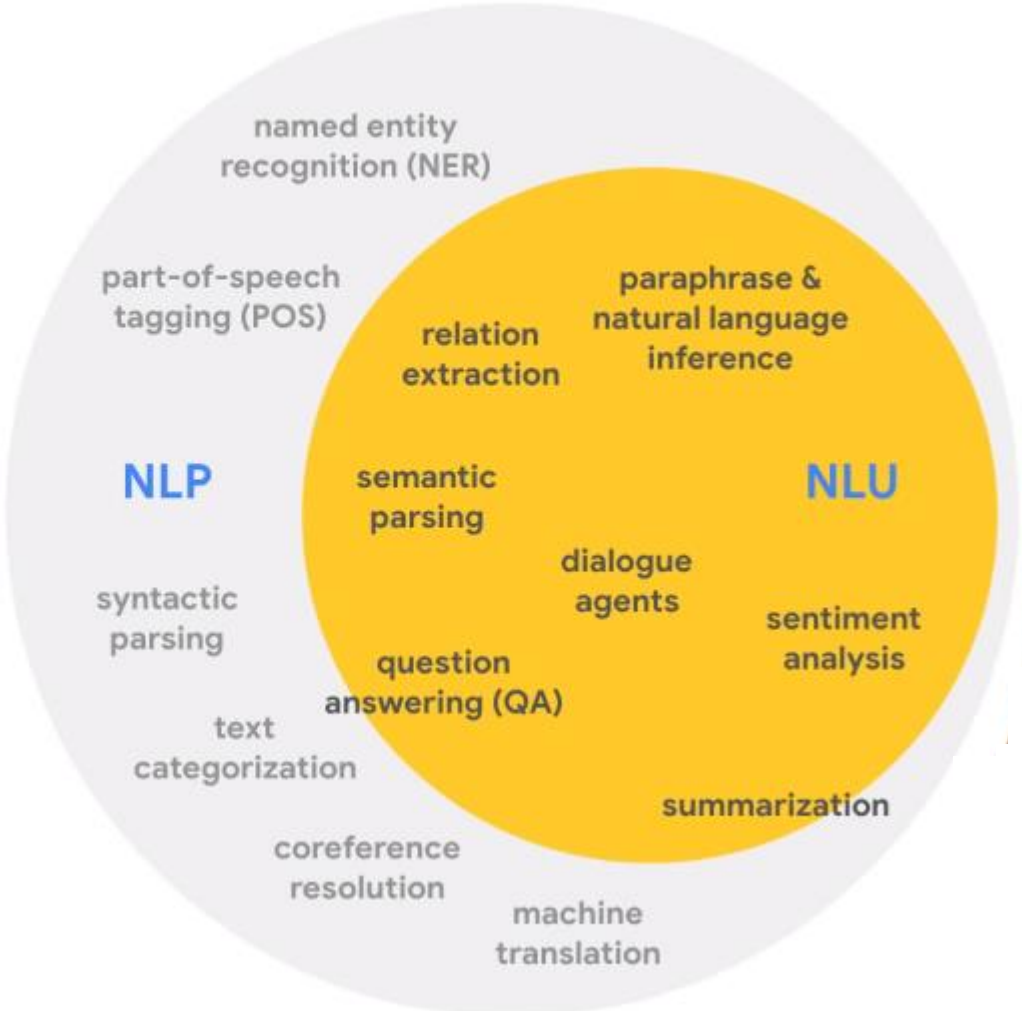
::: the result of an action that the chatbot performed by interacting with one or more backend application

::: a disambiguating question that helps the chatbot to correctly understand the user's request

Bot Engine / Platform

Bot development platform are software frameworks that abstract away much of the manual work that is involved in building chatbots.

Terminology: NLU vs. NLP vs. ASR



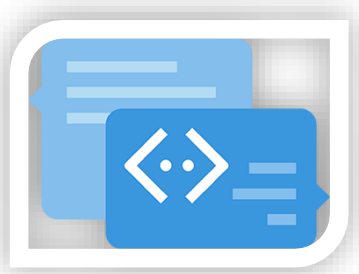
Dialogflow

NLU Engine

Top Bot Platform

Watson Assistant

Microsoft Bot Framework



Dialogflow

Amazon Lex

No Code Bot Platform

ManyChat

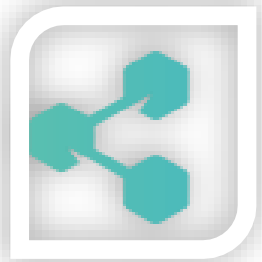
Chatfuel

and more ...

Open Source Bot Platform

Rasa

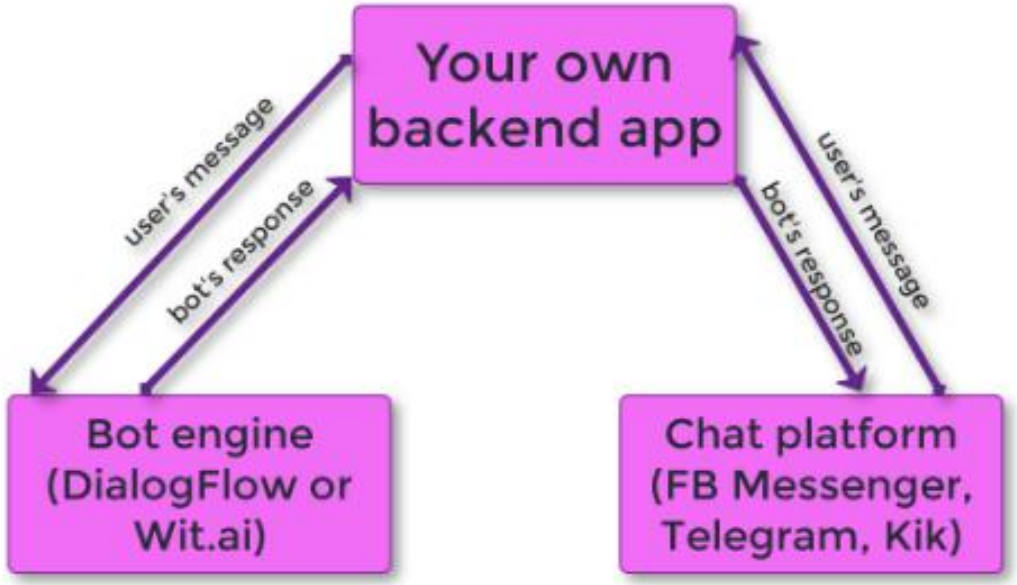
Botpress



and more ...

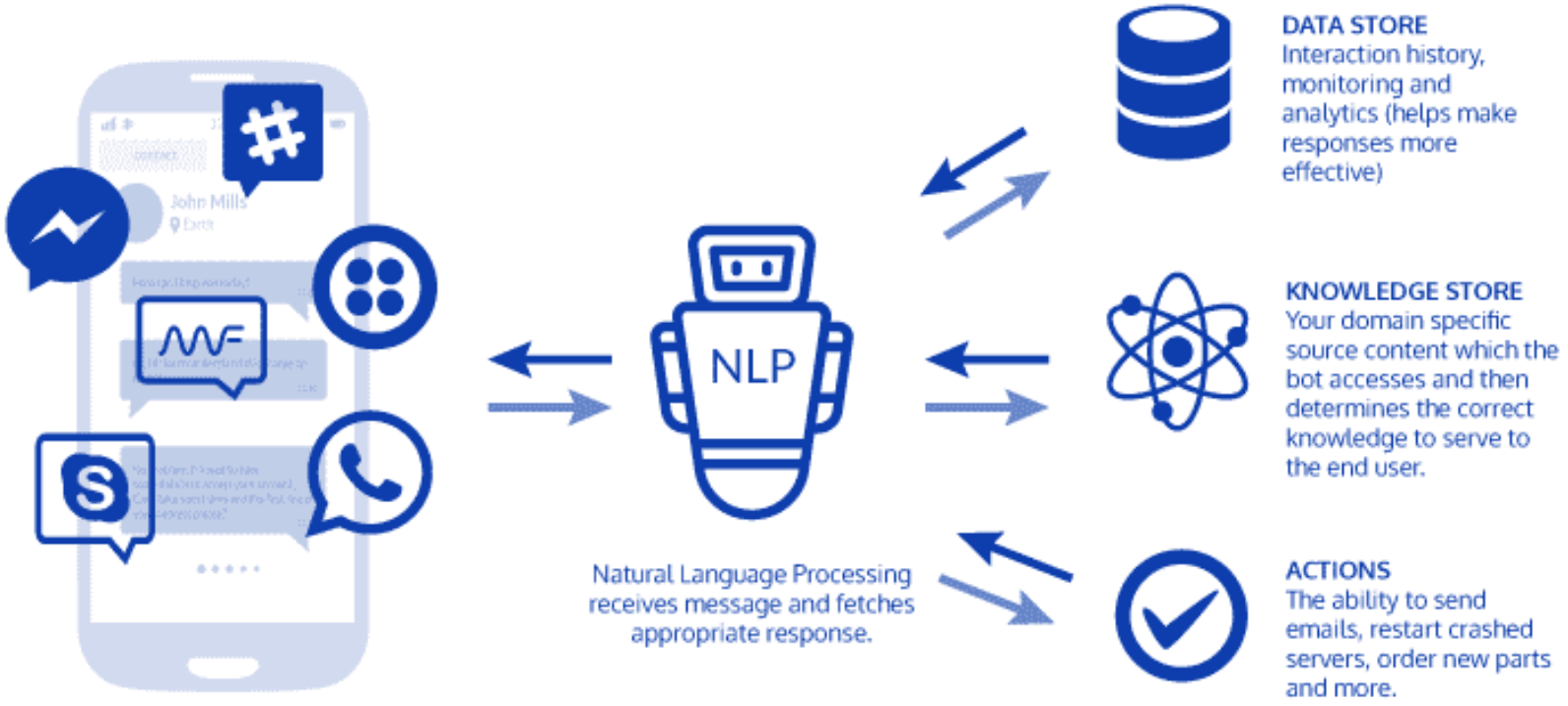
ChatBot Reference Architecture

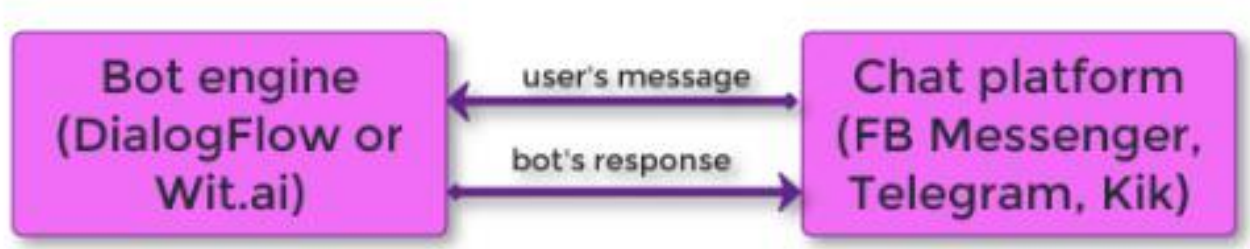
Typical chatbot architecture



ChatBot Reference Architecture

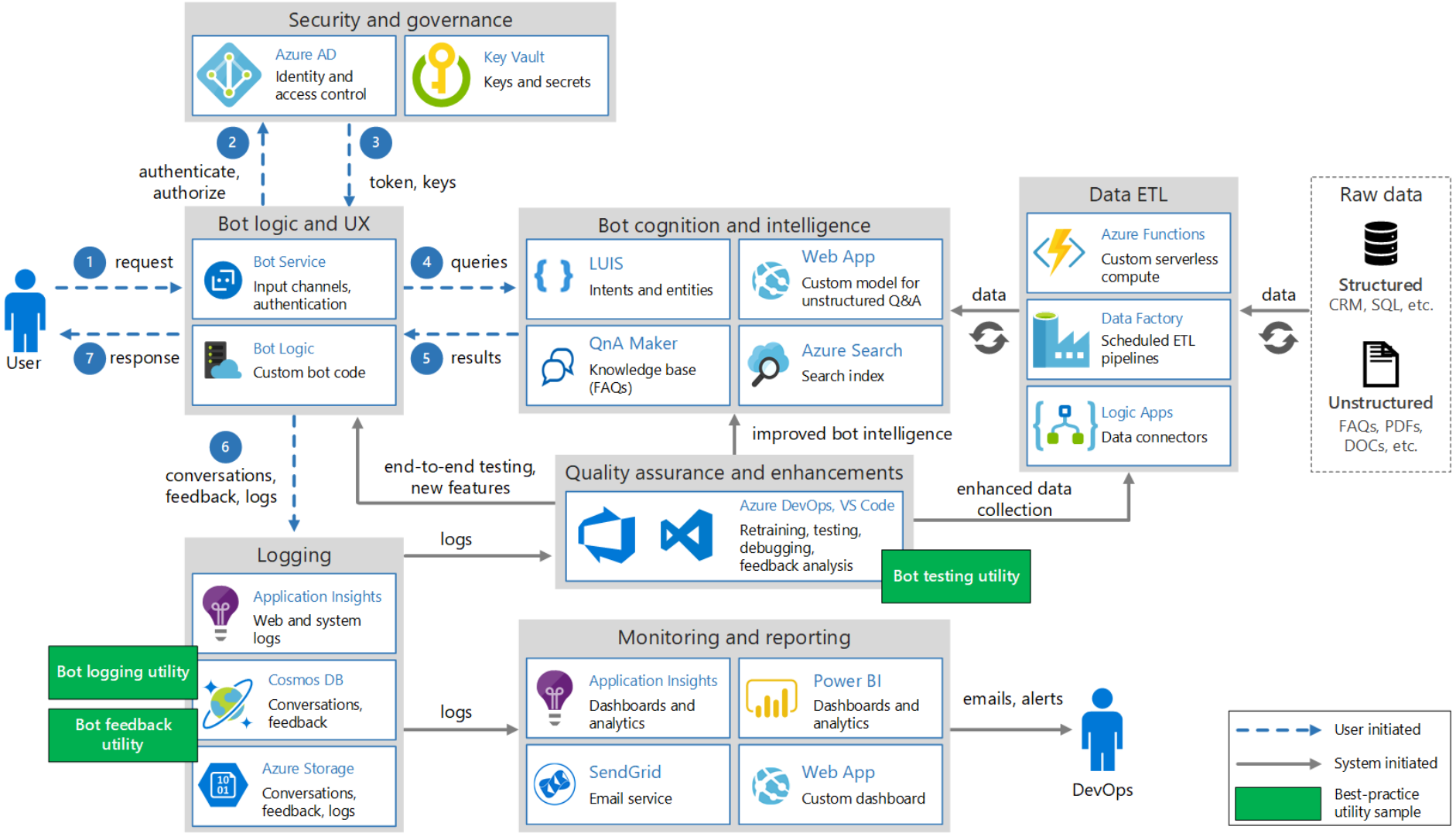
Simple chatbot architecture





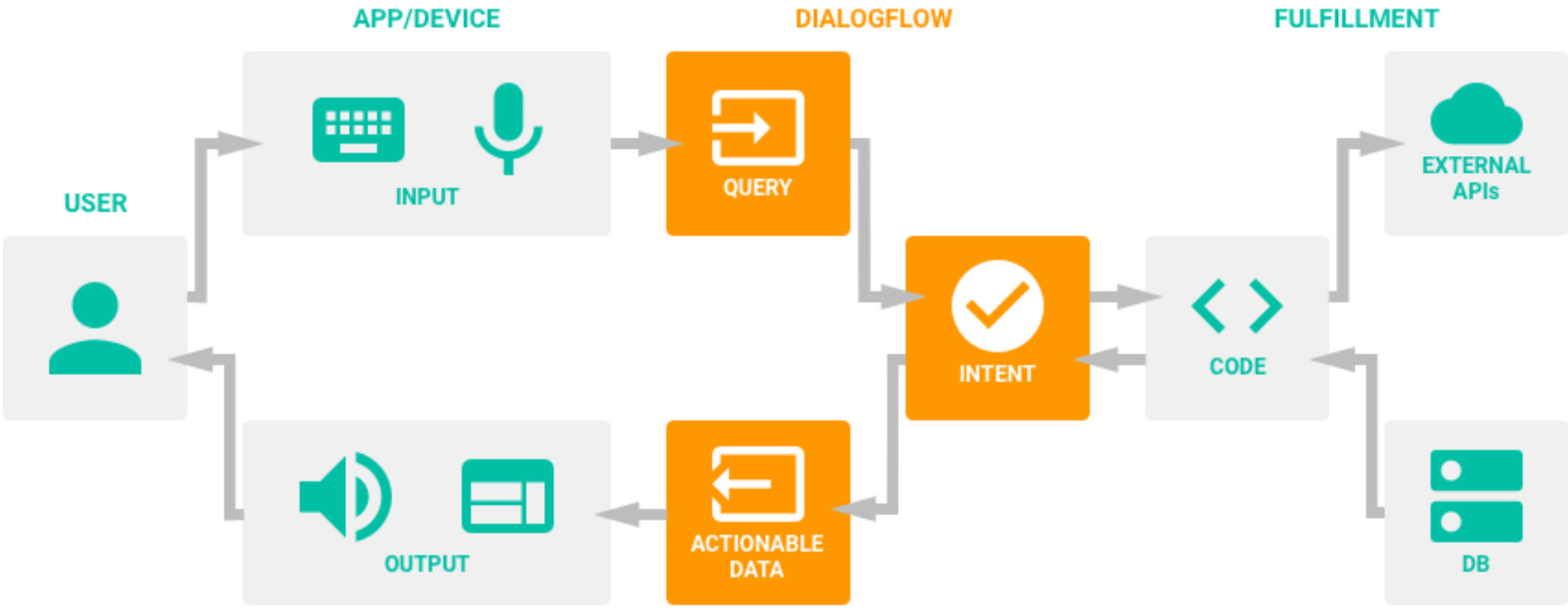
ChatBot Reference Architecture

Simple chatbot architecture (with Azure Platform)



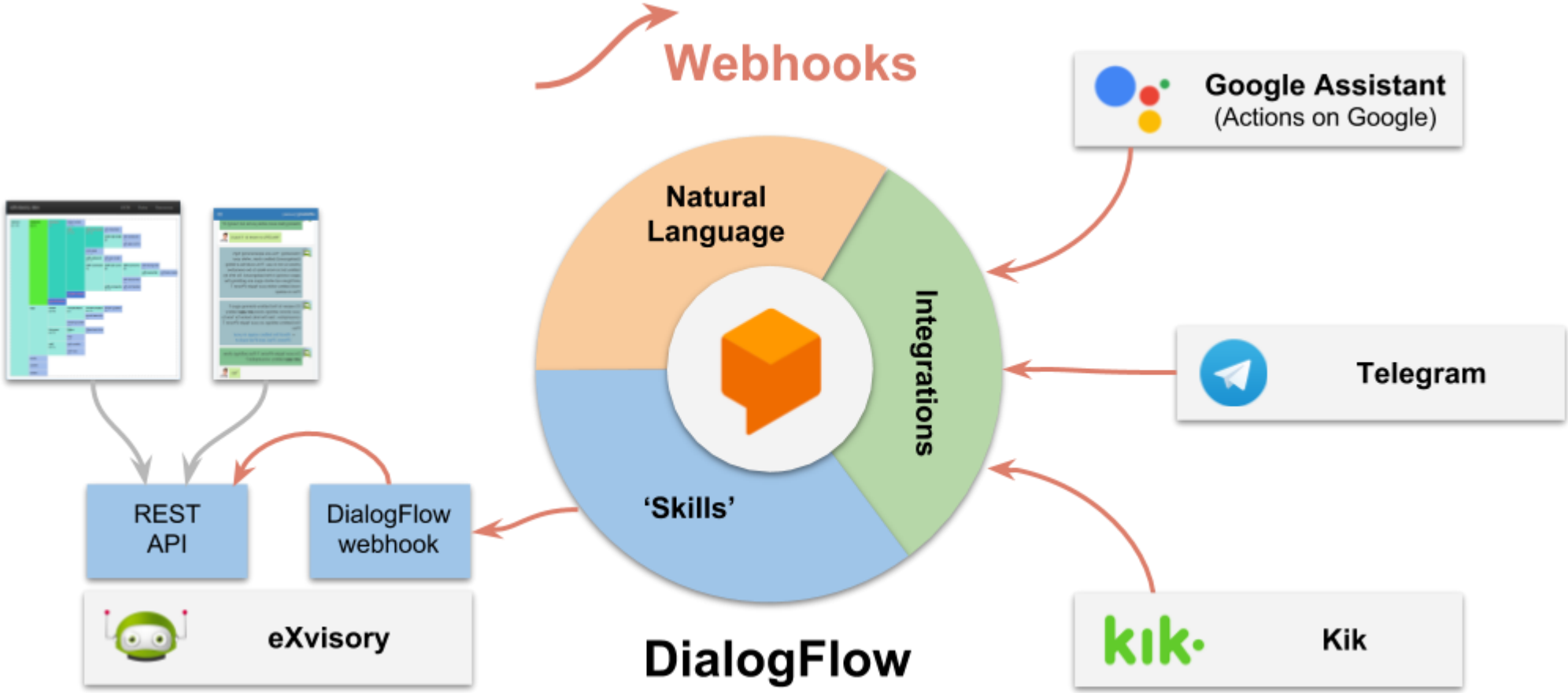
ChatBot Reference Architecture

Simple chatbot architecture with DialogFlow



ChatBot Reference Architecture

Simple chatbot architecture with DialogFlow



Steps to design a ChatBot

1. Determine your bot's purpose
2. Decide between a rule-based and NLP platform
3. Choose Platform
4. Define personality and tone
5. Capture Requirements
6. Integrate visuals
7. Take Analytics into Account

Steps to design a ChatBot

1

Determine your bot's purpose

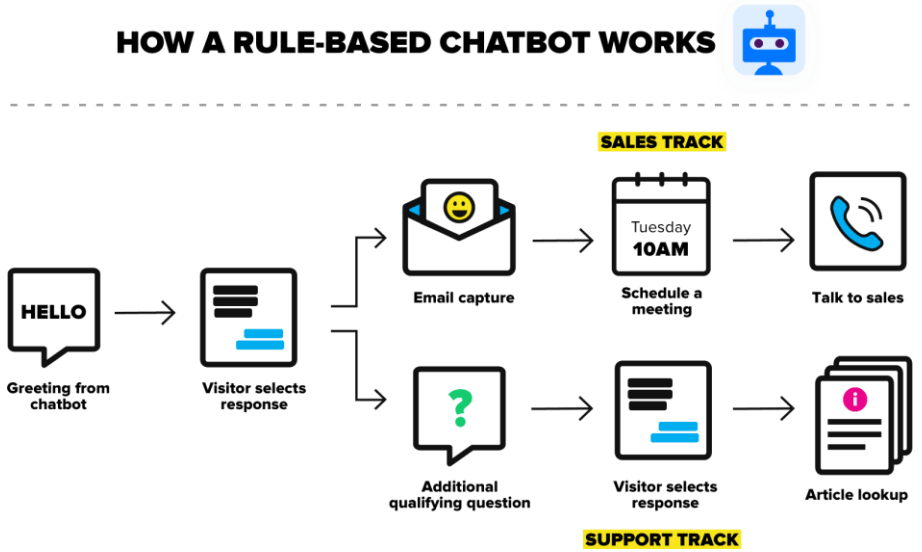
Why do you need a bot? If you cannot answer this question with conviction, then you may want to rethink if you really need one.

> *The users are using your chatbot for one reason, and one reason only: to seek an answer to one of their problems.*

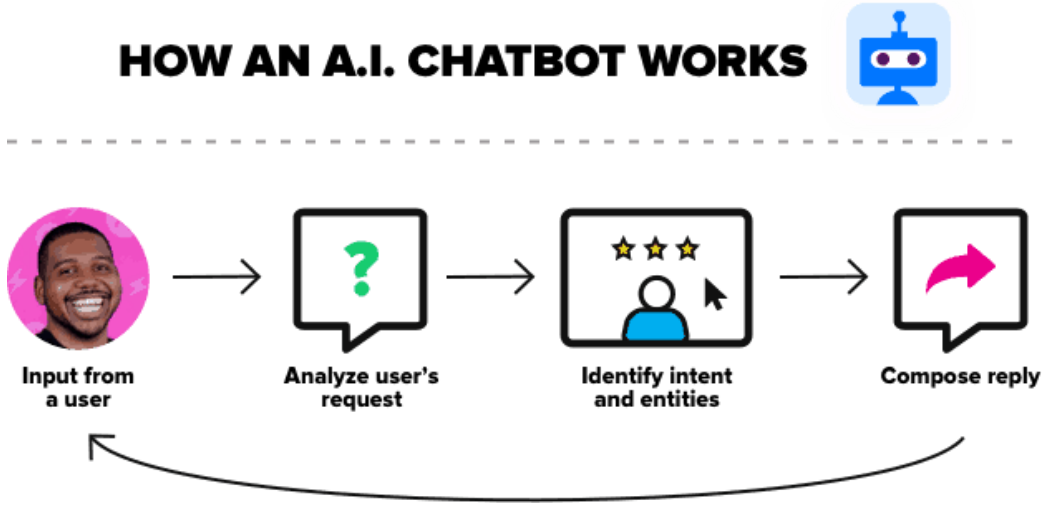
Steps to design a ChatBot

2 Decide between a rule-based and NLP

Rule-based bots chat according to defined decision trees, like a flowchart.



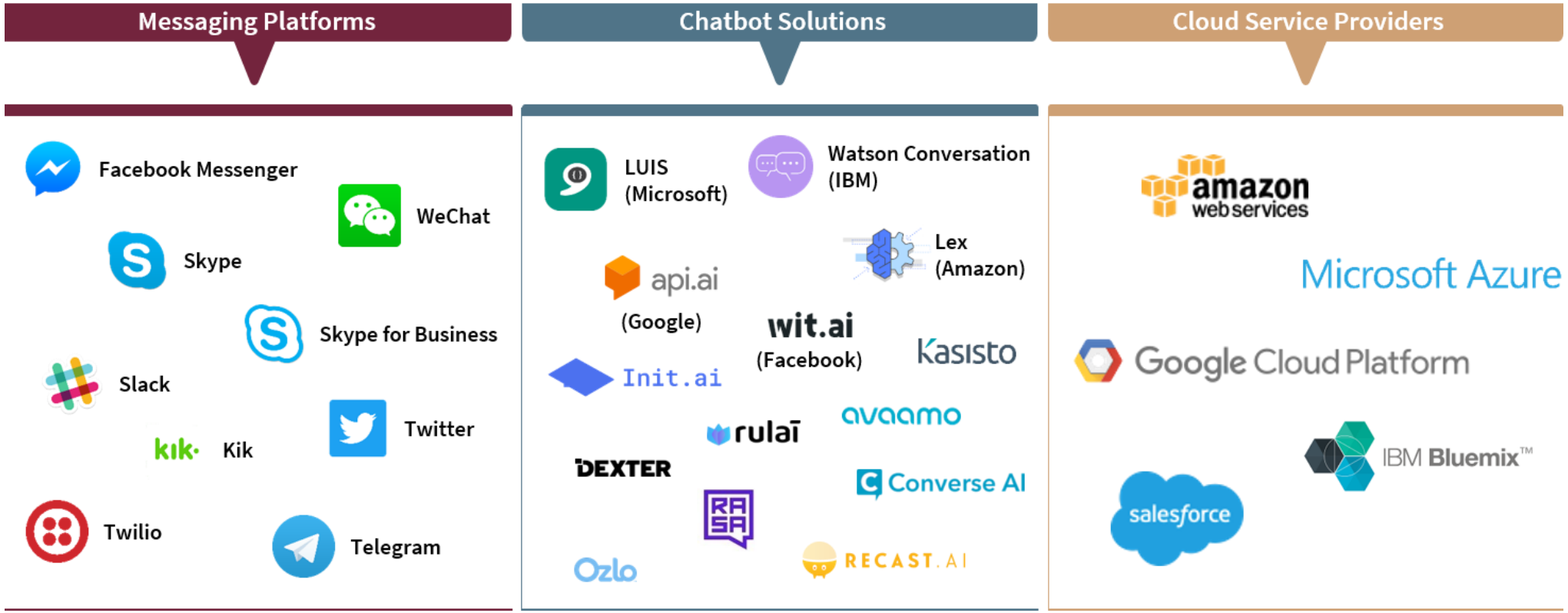
Bots with Natural Language Processing (NLP) are able to understand the context even when questions are more complex.



Steps to design a ChatBot

3

Choose Platform



Steps to design a ChatBot

4

Define personality and tone

Chatbot is an additional way of interaction between your customer and your brand. This is why this experience must be consistent with the other elements of your brand's style.

It's important to design its language in line with your corporate identity.

Steps to design a ChatBot

5

Capture Requirements

If it’s easy to identify the user groups for your chatbot, you can apply a standard framework for user stories.

Understanding your users’ needs, behavior, and expectations is one of the keys to success.

If there are different user types within your brand target auditory, it’s necessary to identify them all from the early start. When it’s done, you can figure out who your bot interacts with and how the bot can enrich relations between these people and your brand.

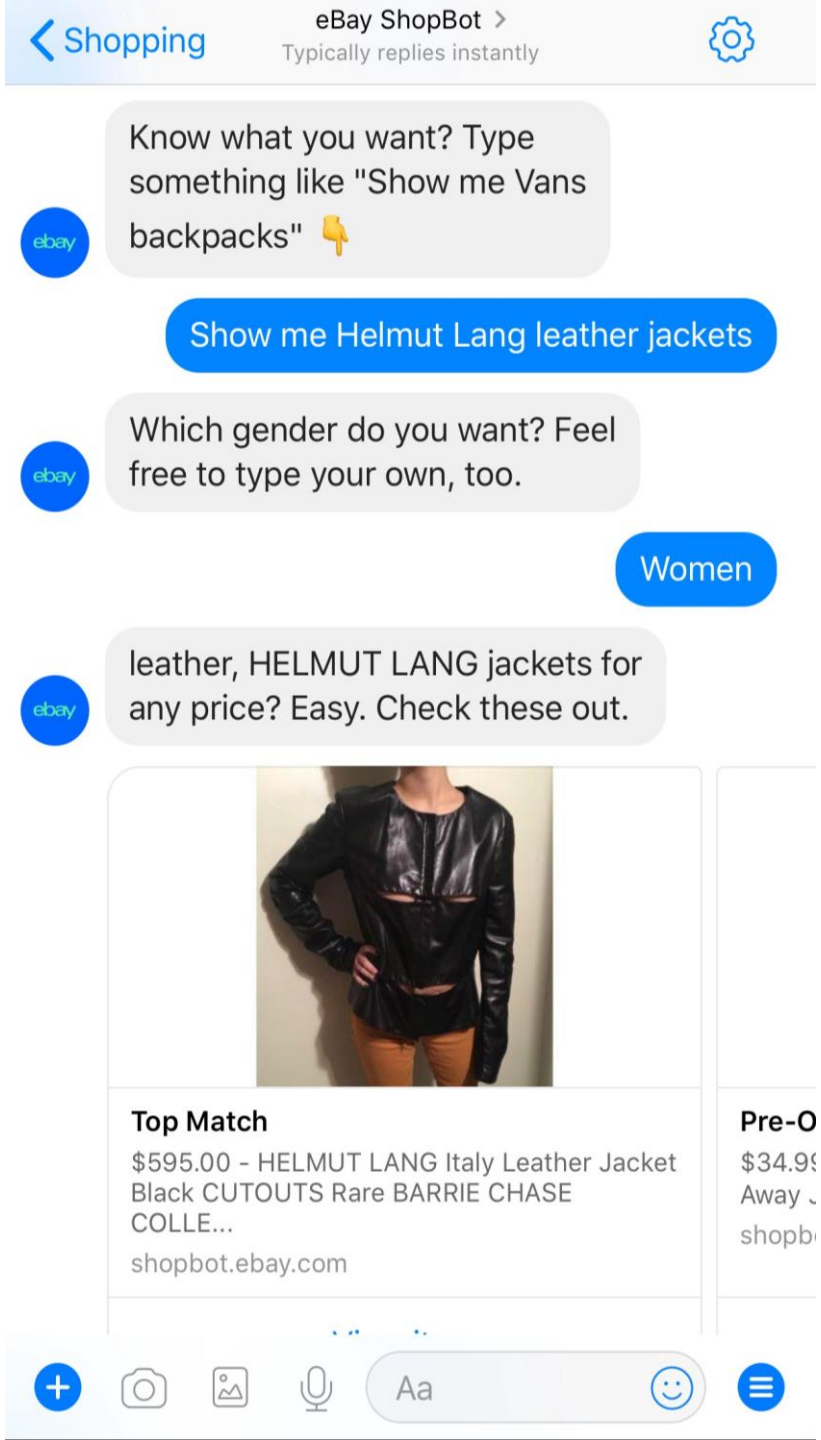
Steps to design a ChatBot

6

Integrate visuals

Besides the text, visuals are the second most important and useful element of your chatbot design.

Visuals are processed 60,000 times faster than text. This means using images to illustrate your chatbot's messages are likely to hook your user's interest.



Steps to design a ChatBot

7

Take Analytics into Account

To monitor its performance, you need to choose a proper tool for analytics. The tool that will help you keep an eye on the way your customers interact with the bot.

Let’s get our hands dirty

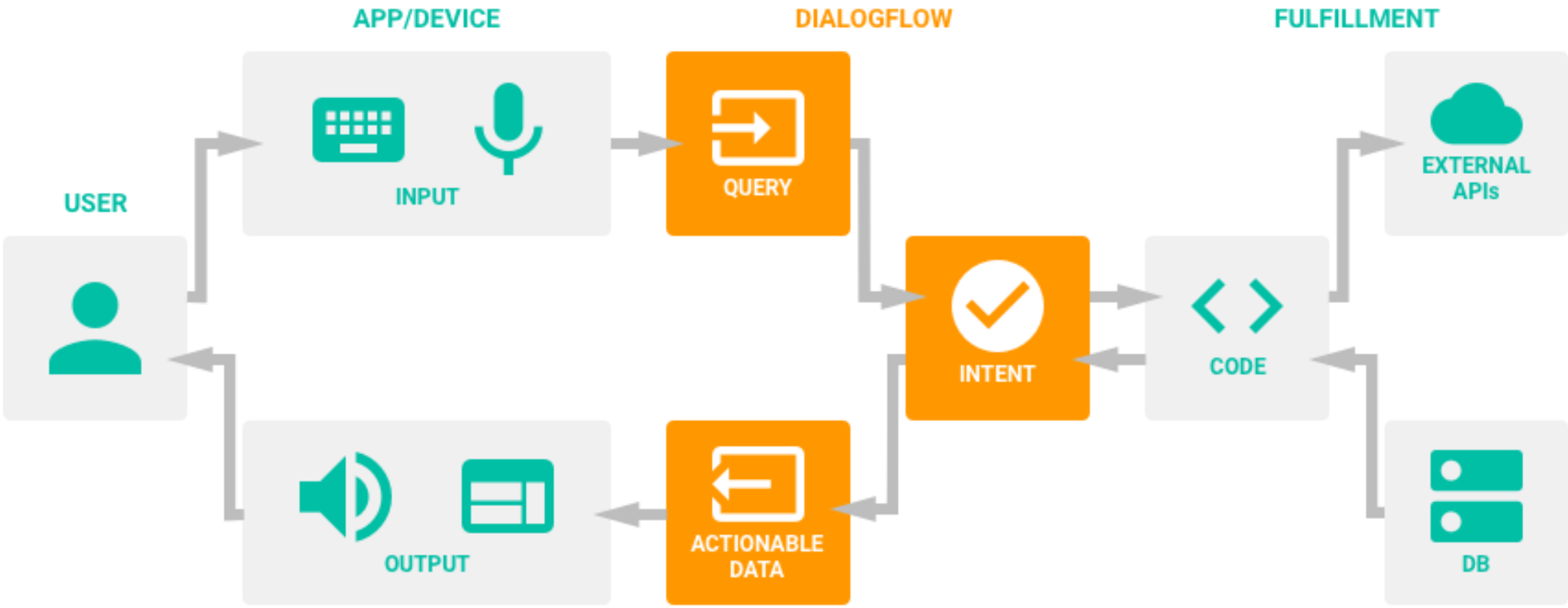
Build an appointment scheduler with Dialogflow

What we'll learn

- ✓ How to create a Dialogflow agent
- ✓ How to create intents in an agent
- ✓ How to create training phrases in an intent
- ✓ How to create responses in an intent
- ✓ How to test a Dialogflow agent
- ✓ How to set up web integration

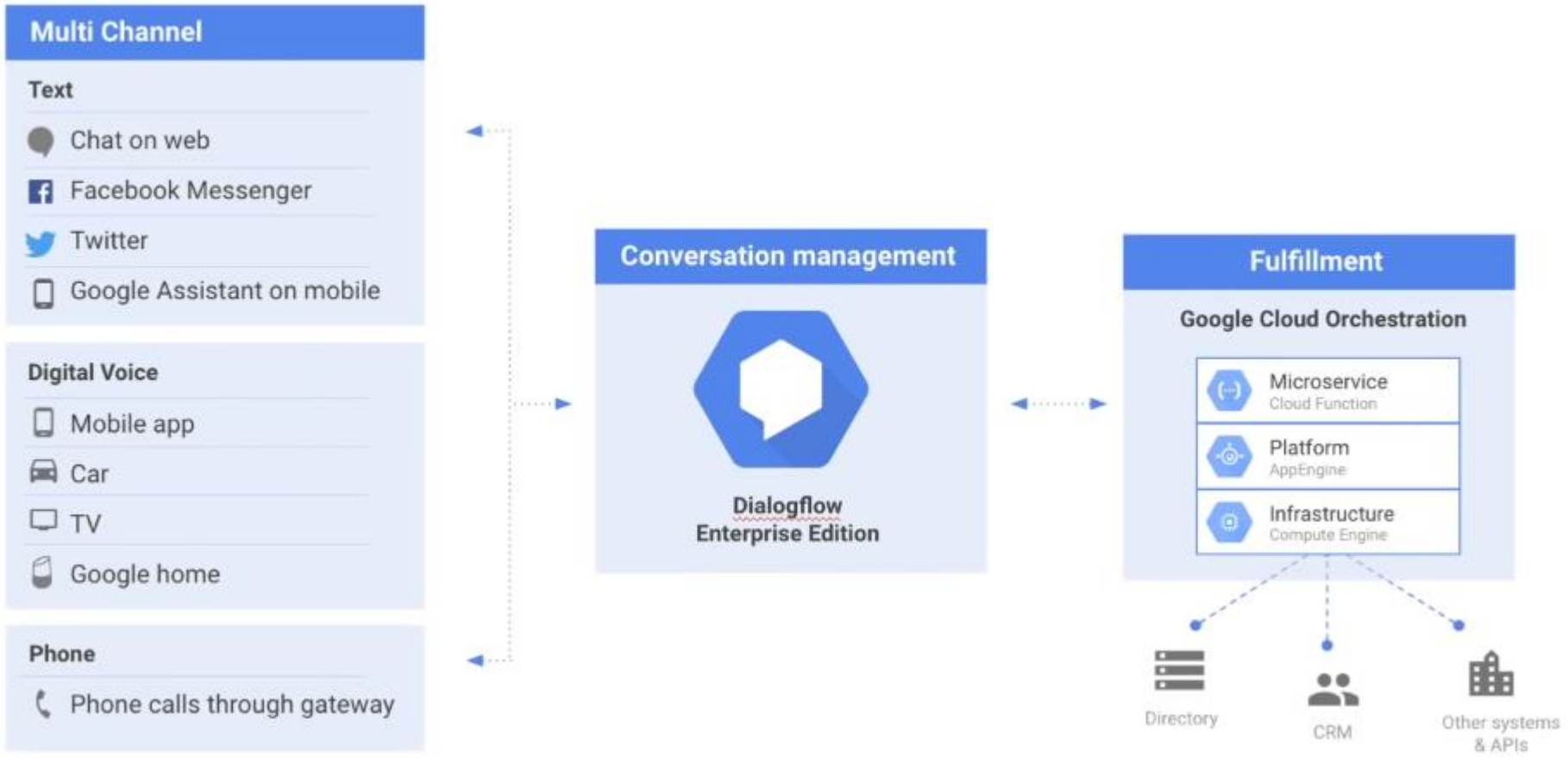
ChatBot Reference Architecture

Simple chatbot architecture with DialogFlow

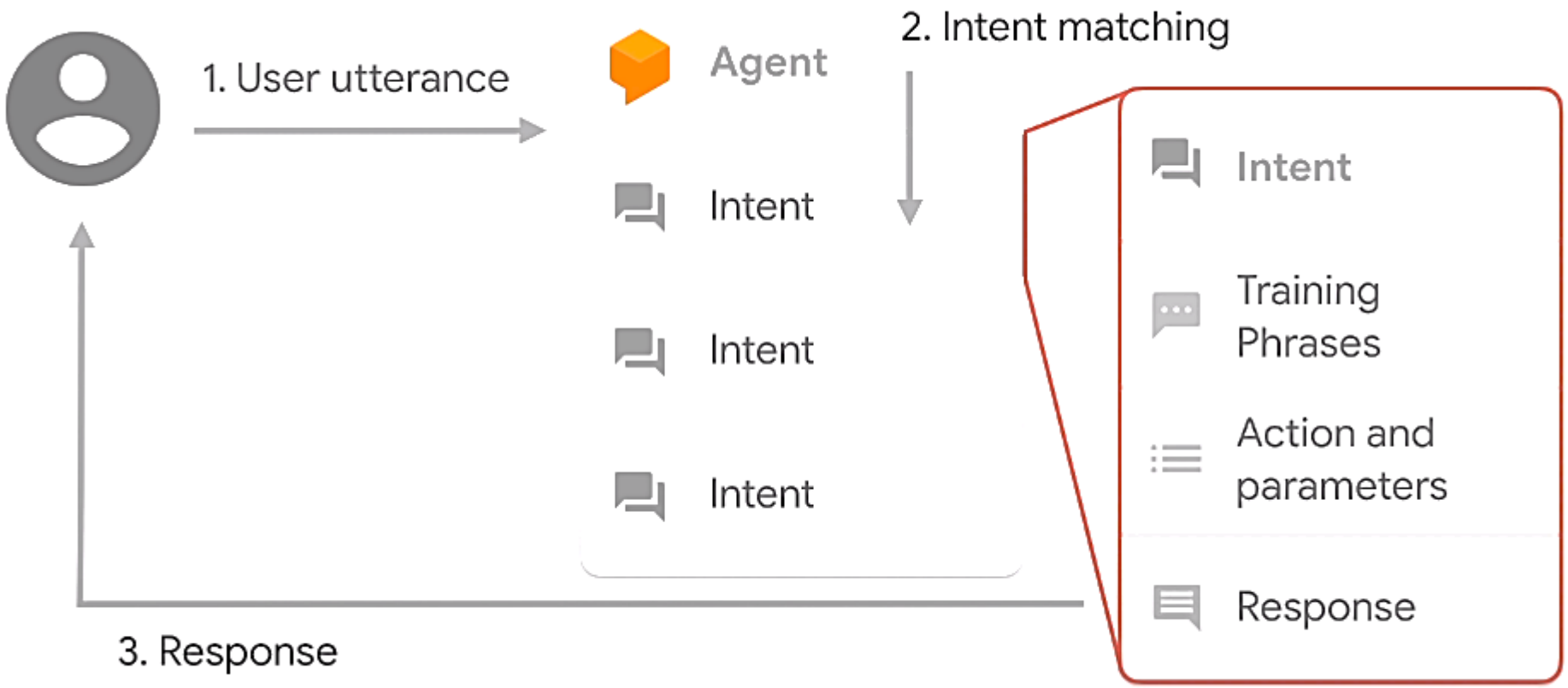


ChatBot Reference Architecture

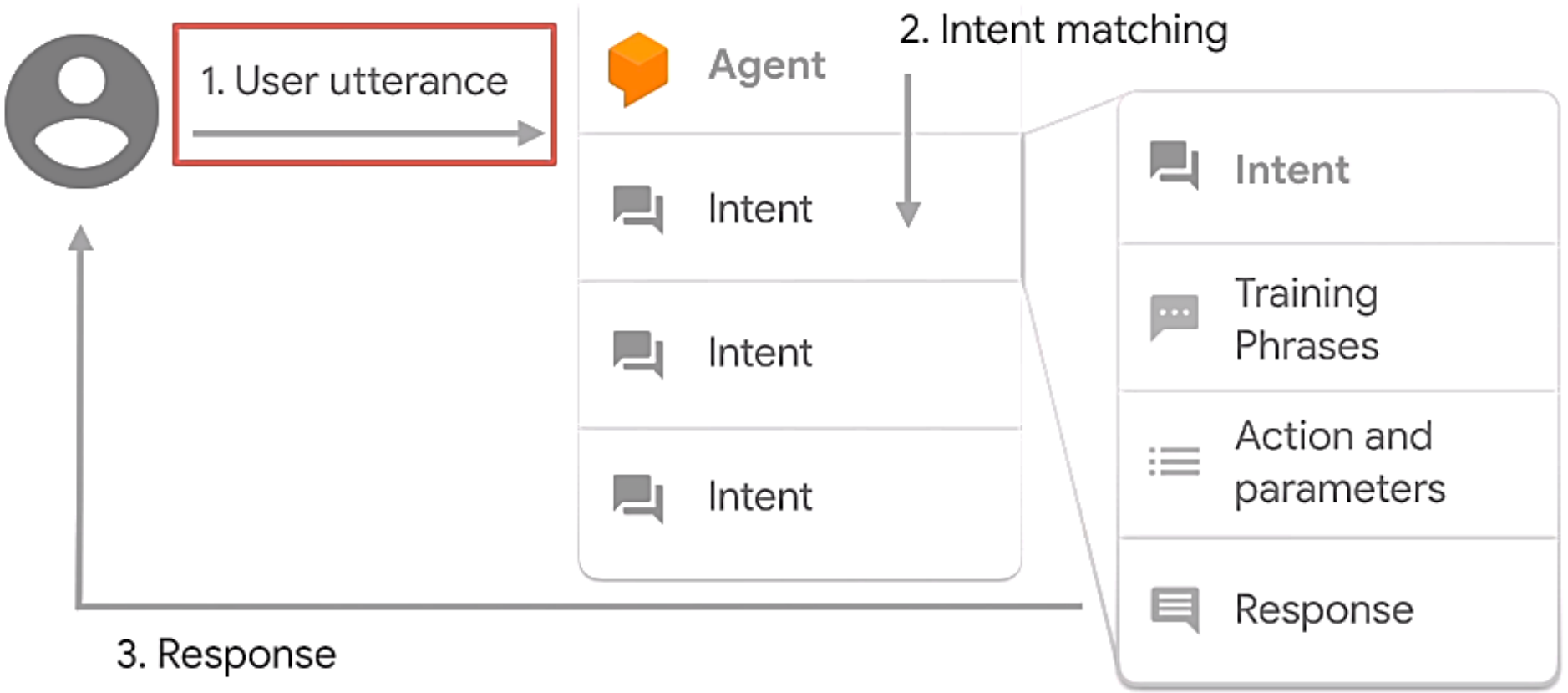
Simple chatbot architecture with DialogFlow



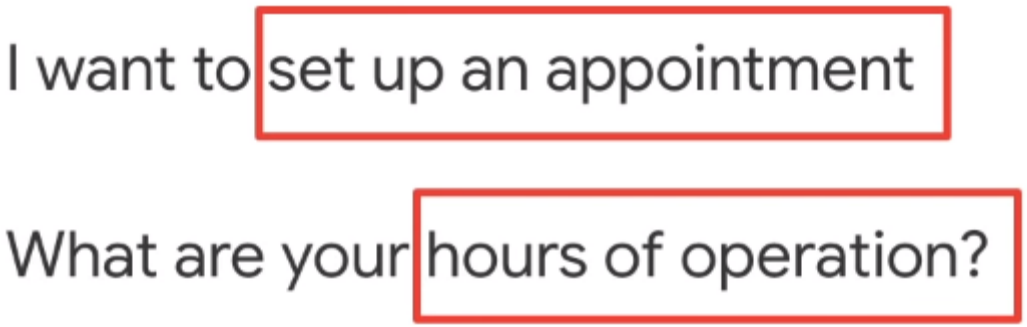
Agent


1. User utterance
Agent
Intent
Intent
Intent
2. Intent matching
Intent
Training Phrases
Action and parameters
Response
3. Response

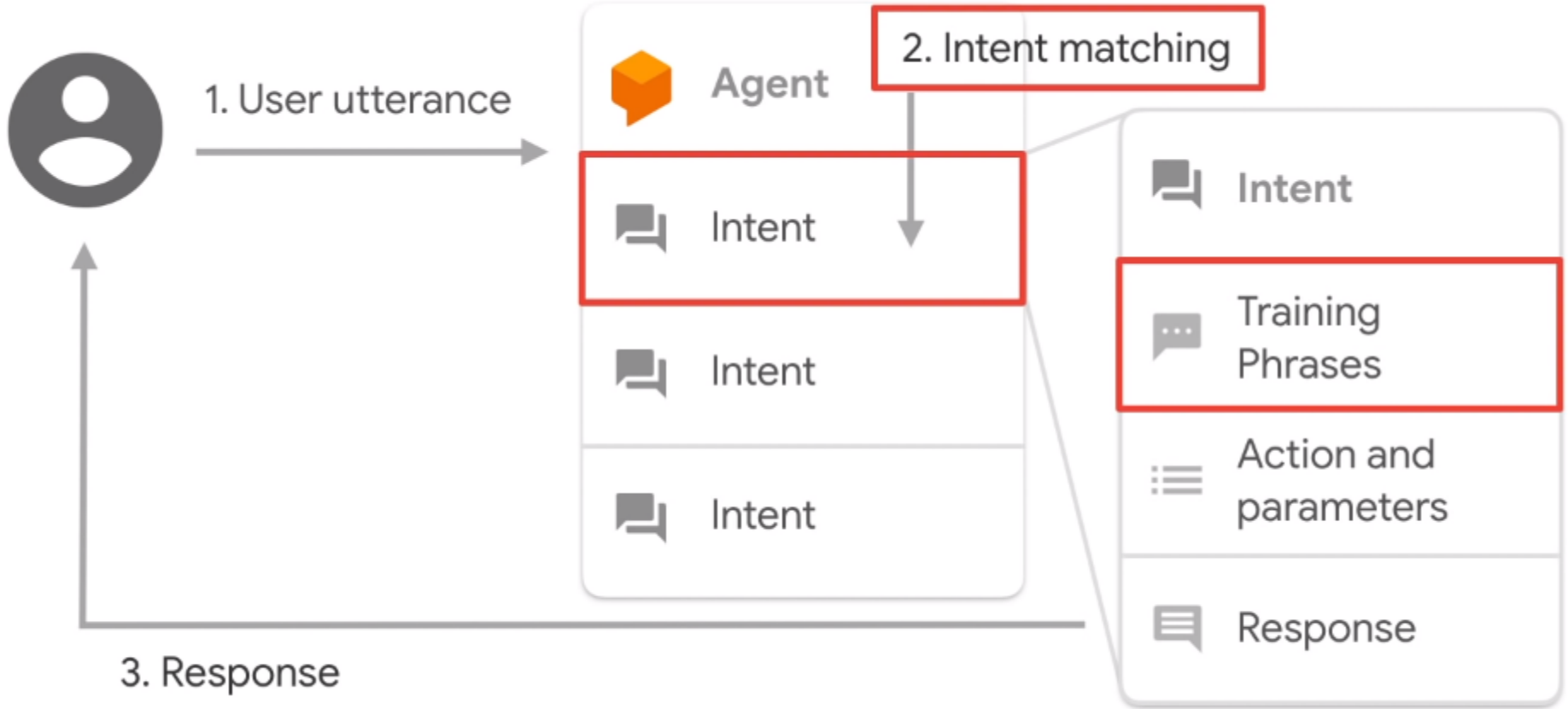
Utterance


1. User utterance
Agent
2. Intent matching
Intent
Intent
Intent
Intent
Training Phrases
Action and parameters
Response
3. Response

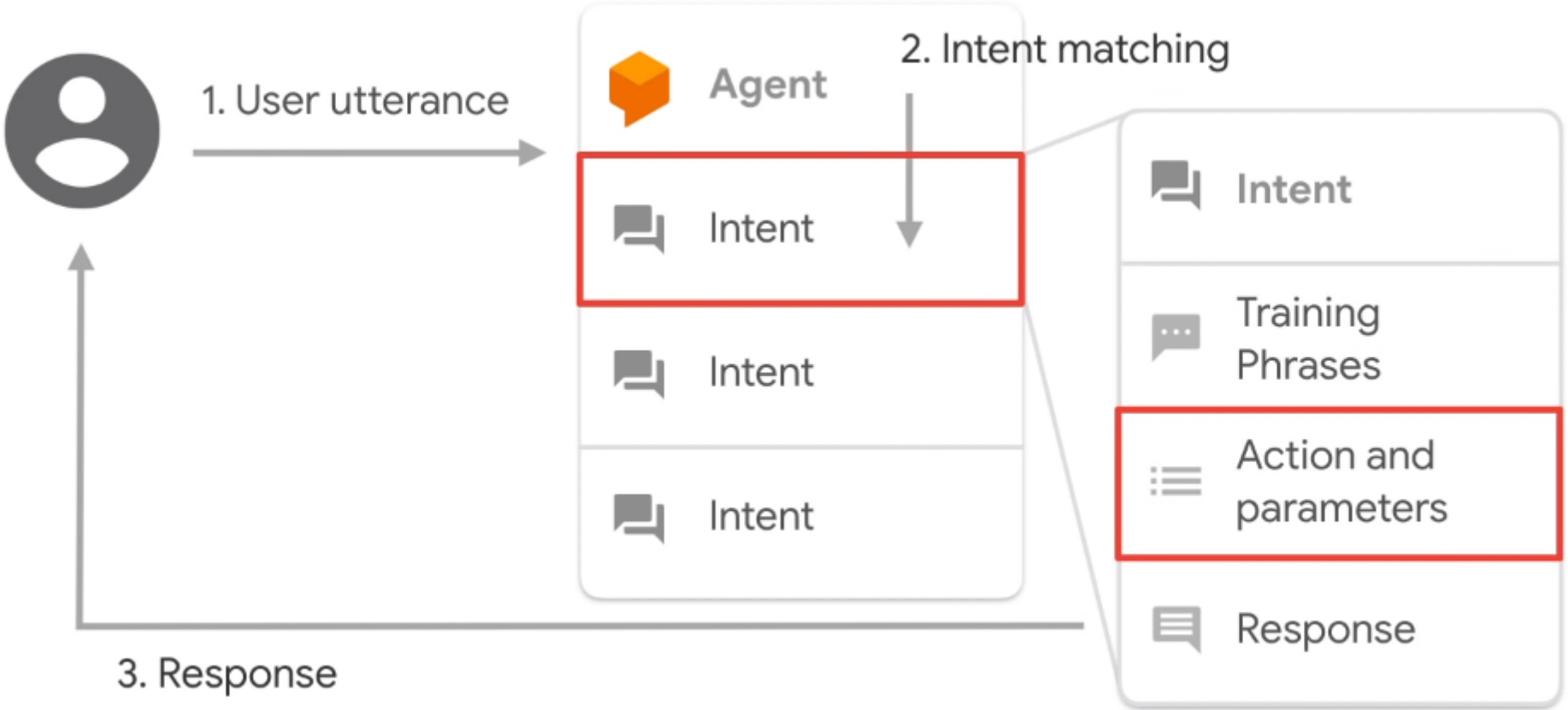
Intent


I want to set up an appointment
What are your hours of operation?

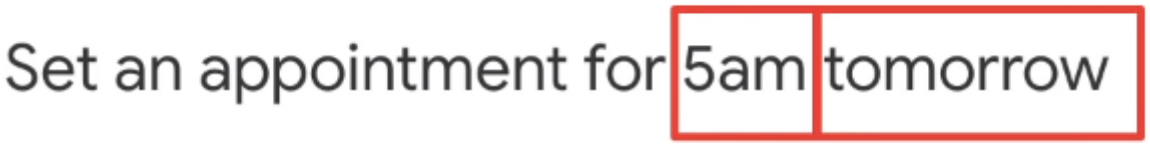
Intent – Training Phases & Intent Matching



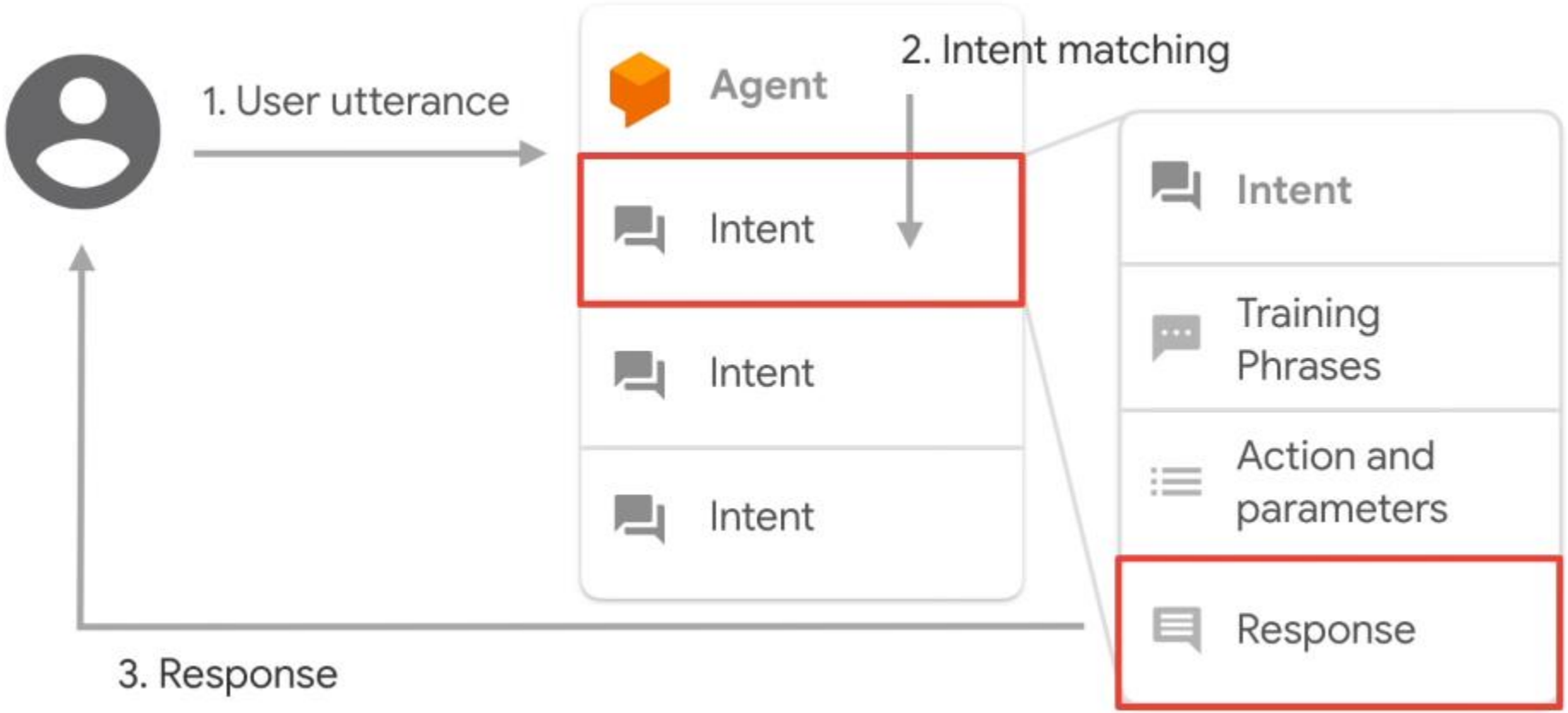
Intent – Action and Parameters



Entities



Intent - Response



Entities

- System Entities
- Developer Entities
- Session Entities

System Entities

Date and time
@sys.time
@sys.date

Numbers
@sys.number
@sys.flight-number

Amounts with units
@sys.unit-currency
@sys.unit-length

Geography
@sys.address
@sys.airport

Contacts
@sys.email
@sys.phone-number

Names
@sys.given-name
@sys.last-name

Developer Entities

@service-option

Tune-up

Repair

Tire change

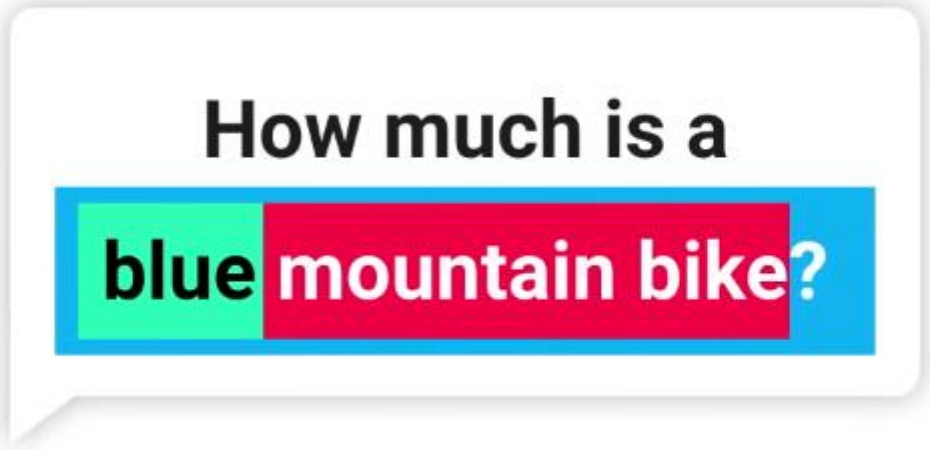
Upgrade

Synonyms

- service, tuning, tune
- fix, mend, restore, overhaul
- puncture repair
- improve, update, modernize

Developer Entities

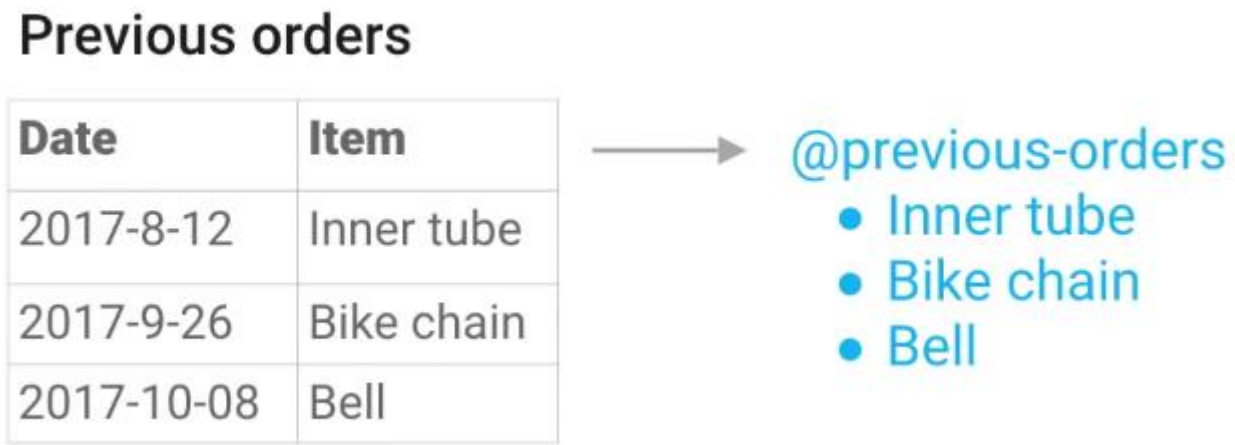
@bike-with-color
- @sys.color @bike-type



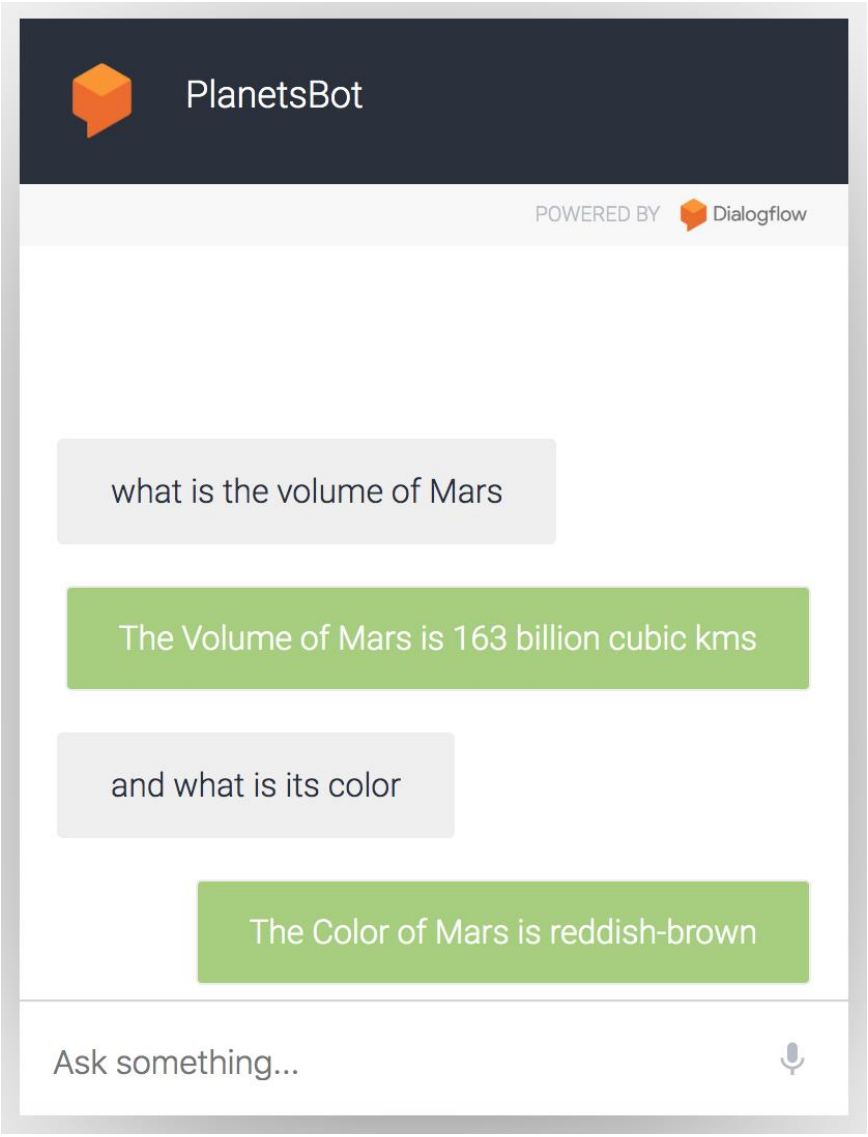
blue mountain bike
- blue
- mountain bike

@bike-with-color
- @sys.color
- @bike-type

Session Entities



Context


PlanetsBot
POWERED BY Dialogflow
what is the volume of Mars
The Volume of Mars is 163 billion cubic kms
and what is its color
The Color of Mars is reddish-brown
Ask something...

Webhook

```
let responseJson = {};
responseJson.fulfillmentText = 'This is an endpoint published to RunKit'; // displayed response
if(action === 'getPlanetAttribute'){
    let planetName = req.body.queryResult.parameters.planet;
    let attributeName = req.body.queryResult.parameters.attribute;

    var request = require("request" 2.88.0 );

    let responseJson = {};
    responseJson.fulfillmentText = 'This is an endpoint published to RunKit'; // displayed response
    base('PlanetsTable').select({
        maxRecords: 1,
        filterByFormula: 'AND({Planet} = "'+planetName+'",{Attribute}="'+attributeName+'")',
    }).eachPage(function page(records, fetchNextPage) {
        records.forEach(function(record) {
            console.log('Retrieved', record.get('Value'));
            responseJson.fulfillmentText = 'The ' + attributeName+' of '+ planetName+ ' is '+record.get('Value
            res.json(responseJson);
        });
        fetchNextPage();
    }, function done(err) {
        if (err) { console.error(err); return; }
        res.send(resp);
    });
}
else if(action === 'changedAttribute'){
    //let planetName = req.body.queryResult.parameters.planet;
    let attributeName = req.body.queryResult.parameters.attribute;
```

For the getPlanetAttribute action we run this code block